

CD-doc-2345

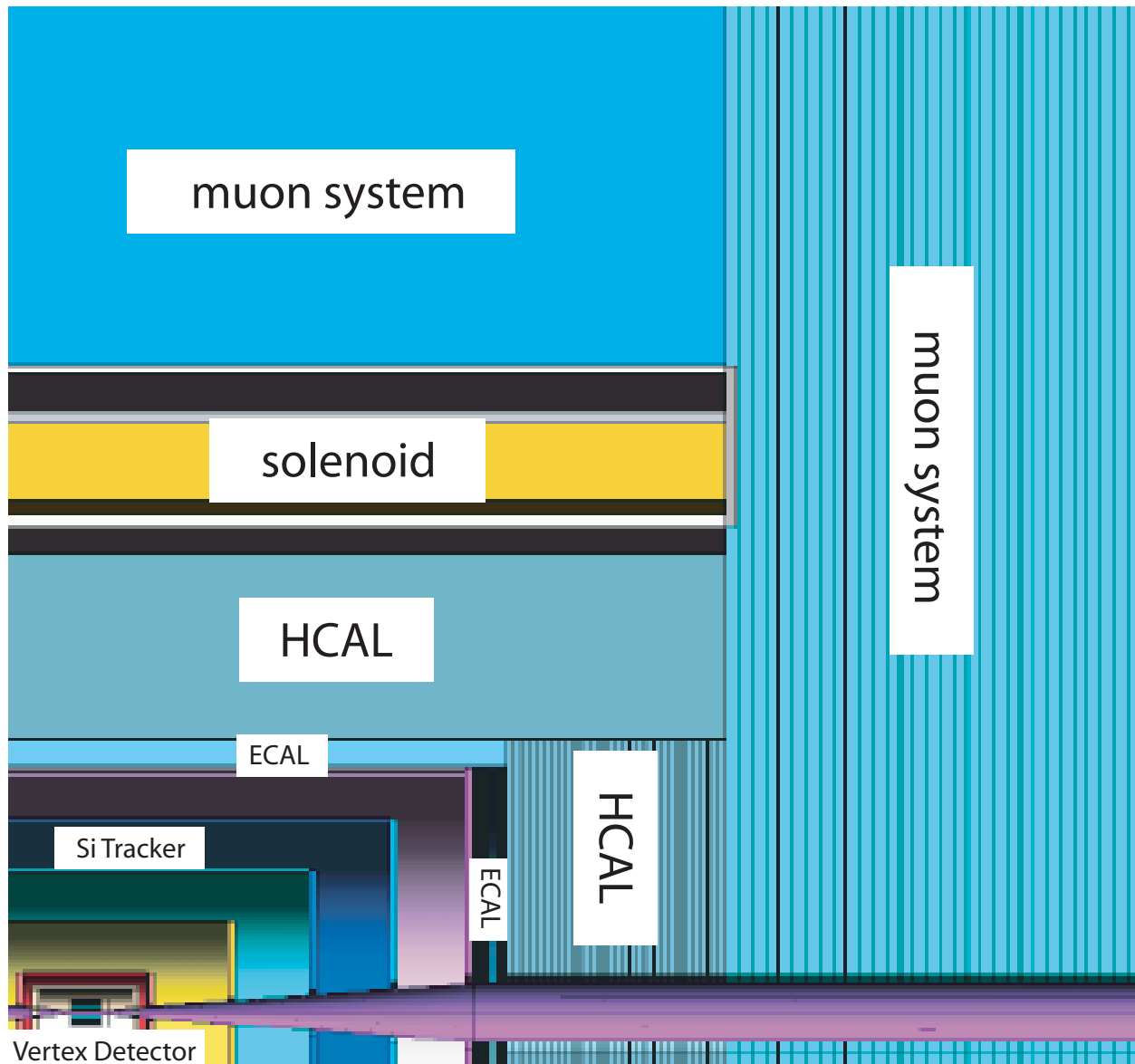
SiD Simulations and Benchmarking

Rob Kutschke, CD/IDS
ILC Coordination Forum
July 17, 2007

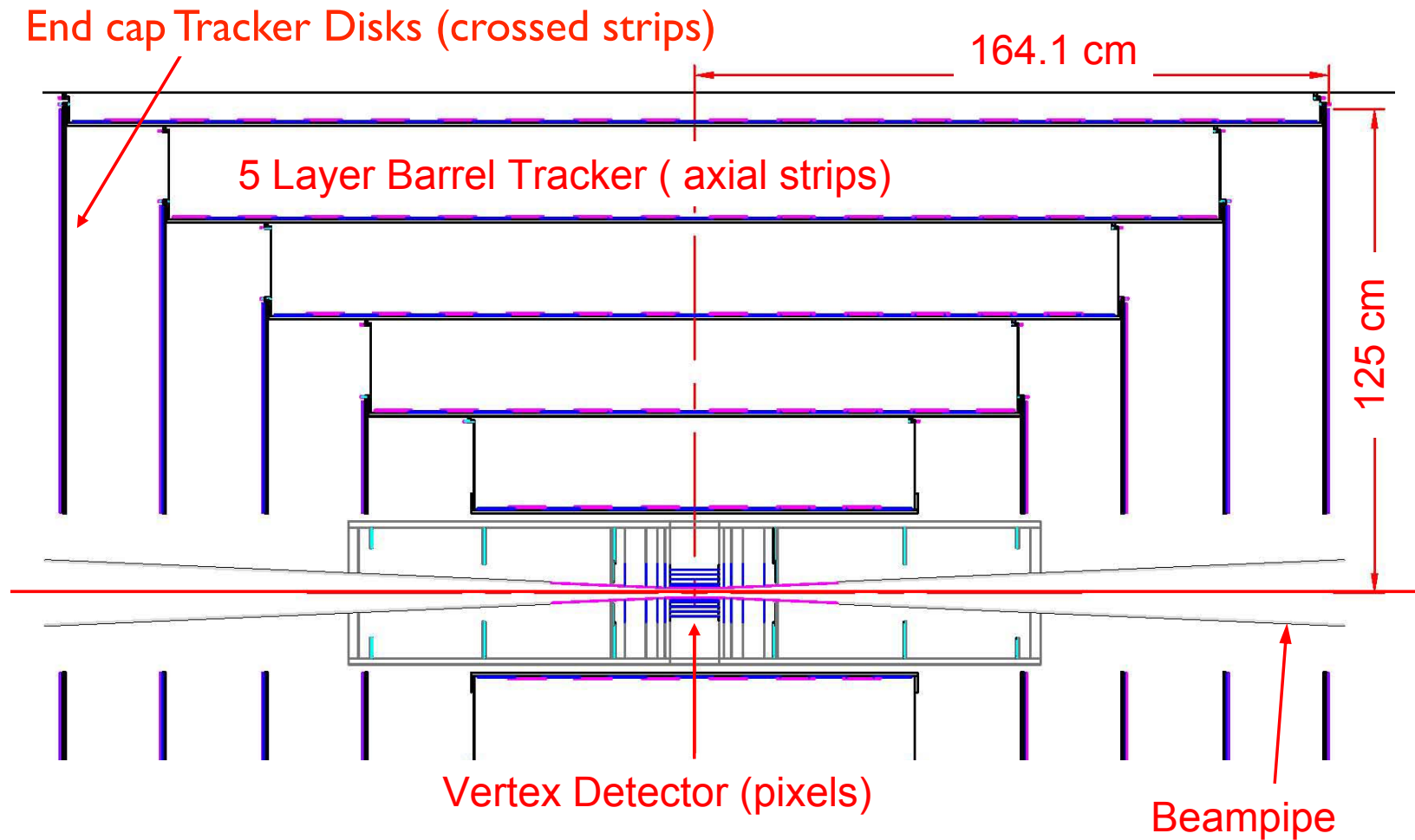
Outline of This Talk

- SiD Detector
- SiD Software
- Work plan
 - Overview of all planned work
 - Summary of Rob's work
 - Hans' talk will give more details on
 - The forward tracking work
 - Simulation of Pb Glass-scintillator dual readout calorimeter
 - Deadlines

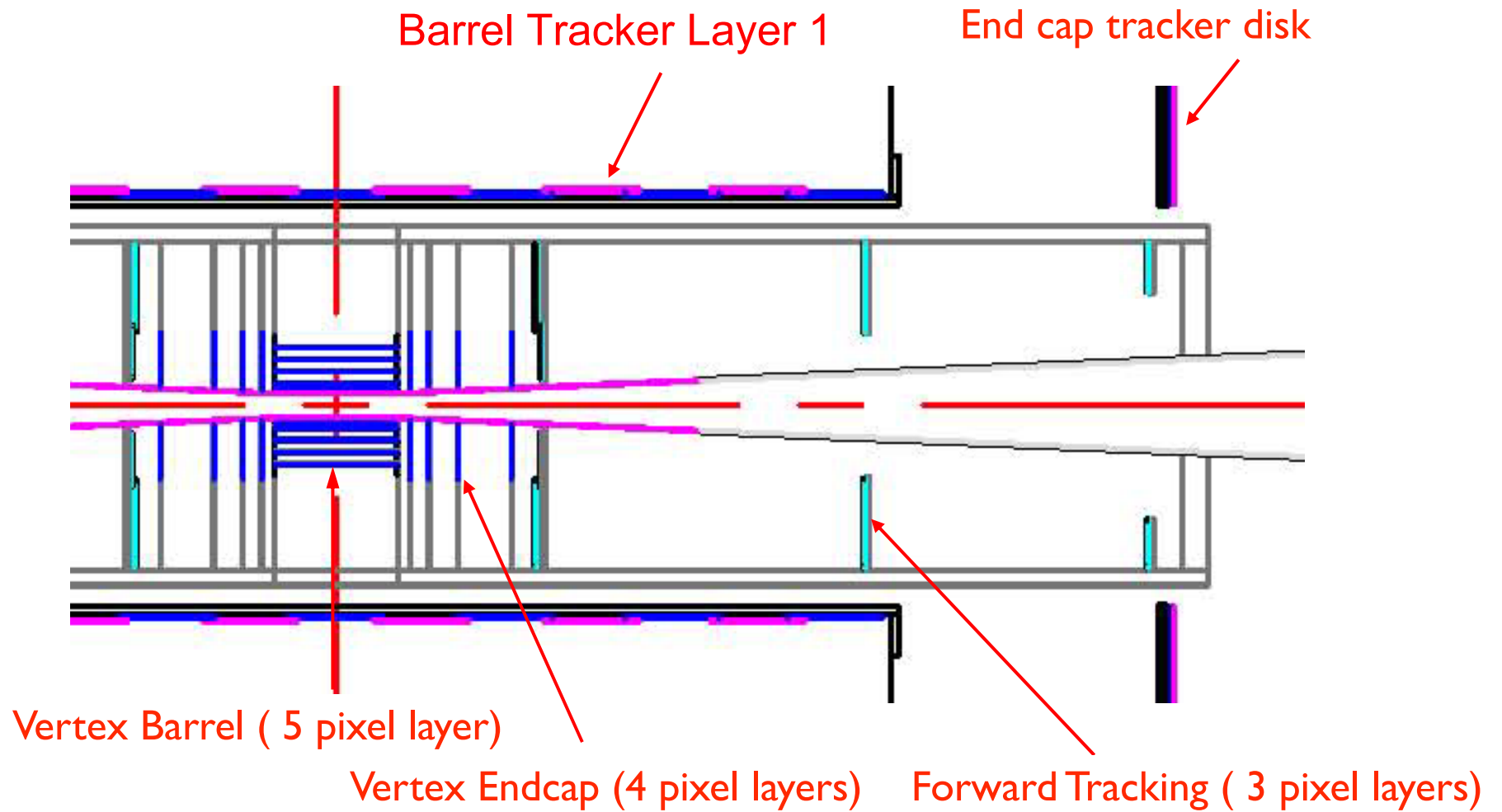
The SiD detector



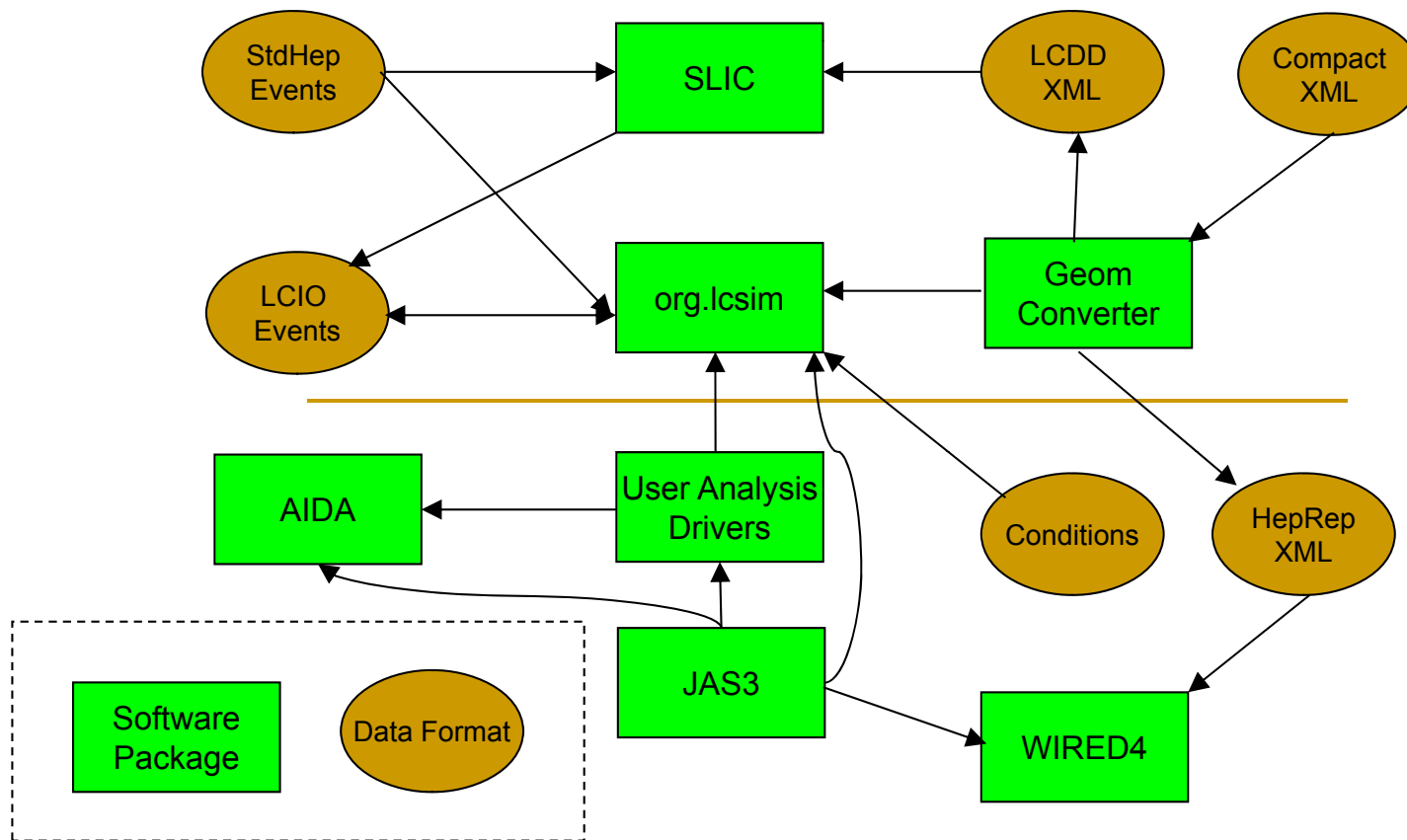
SiD Tracking System, Vertical Section



Detail Near Beam Line



SiD Software Overview



SiD Software

- SLIC:
 - C++. G4 based simulation system.
- org.lcsim
 - Java based framework for reconstruction and analysis.
- Wired 4 based event display.
- GeomConverter:
 - Reads “Compact Detector Description” XML.
 - Native format for org.lcsim.
 - Can write:
 - HepRep XML for Wired-4
 - LCDD XML for SLIC
- Data formats: StdHep and LCIO.

org.lcsim

- Java based.
- Not a full featured framework.
 - Good enough for a small group with documentation by lunch, coffee and beer:30.
- All key reconstruction codes live in user areas.
 - A loose collection of disconnected tools.
 - Historically user codes do not play well together.
 - Hit and Track classes are deficient so everyone makes private extentions.
- Native histogram/tuple environment:
 - aida + JAS3 as a viewer; much poorer than root.
- Weak release/distribution model.
- More details in backup pages.

SiD Detector Models

- sid00
 - Complete but simplified sensitive volumes.
 - Barrel vertexer and tracker are pure cylinders.
 - Endcaps are annulus of disk.
 - Lots of existing MC needed by PFA people.
- sid01
 - As above but more detail of dead material.
 - Added forward tracker.
 - Current official model.
- New model under development.
 - Tracking elements made from wafers.
 - Will need several variations, especially in forward region, including variations of dead material.

Work Plan

- Forward Tracking:
 - Study occupancies using existing detector models.
 - Help to define the new detector model
 - Includes dead materials
 - Real track reconstruction in forward region.
 - Includes pattern recognition and fitting in presence of backgrounds.
- Simulated analyses.
- SiD at FNAL web site.
- Simulate the Pb Glass-scintillator cal.

Plus whatever infrastructure work is implied

Work Plan - Next Level of Detail

1. Study occupancies, using existing models.
2. Bookkeeping and Infrastructure improvements:
 - Help to define new detector model.
 - Details on next page.
3. Get Kalman filter working as a final fitter.
4. Enhancements to org.lcsim
 - Real pattern recognition in forward region.
 - “Port”/exercise existing code:
 - Vertexing/Jet Finding/Jet Flavor Id/
5. Simulated Analyses
 - $B(H \rightarrow b \bar{b})$ and $B(H \rightarrow c \bar{c})$.
6. FNAL web site.
7. Simulate Pb Glass-scintillator calorimeter

Bookkeeping and Infrastructure

- Classes that need to be fixed:
 - RawTrackerHit (sort of a digi)
 - Track
- New classes needed:
 - Clusters of digis and clustering algorithms.
 - Bookkeeping of used hits.
 - Collection of muon and electron candidates.
- We are waiting on code to create RawTrackerHits from SimTrackerHits (create digis from hits).
- We can create classes but not persist them!
 - Agitate for a new persistency model.
- Effort slowed by demand that all persistent classes be usable by all detector concepts.

Work Plan - Who is doing What

1. Study occupancies, using existing models.
 - Fransisco supervised by Hans.
2. Bookkeeping and Infrastructure improvements:
 - Help to define new detector model.
 - Geometry back end being done at SLAC.
 - Hans and students: work with Bill Cooper (PPD) for models of support and variations on the wafer layout.
 - Hit, Track and e/mu classes:
 - Rob and Hans with input from SLAC and others.
3. Get Kalman filter code working.
 - Rob

Work Plan - Who is doing What

4. Enhancements to org.lcsim

- Real pattern recognition in forward region.
 - Hans. This is by far the biggest job.
- “Port”/exercise existing code:
 - Vertexing/Jet Finding/Jet Flavor Id/
 - Rob

5. Simulated Analyses

- $B(H \rightarrow b \bar{b})$ and $B(H \rightarrow c \bar{c})$.
- Rob. Will be done in several iterations.

6. FNAL web site.

- Lynn and Rob

7. Simulation of Pb Glass-scintillator dual readout cal.

- Implementation is almost complete. Hans.
- Adam will take it from there.

Web Site

- ilc.fnal.gov undergoing redesign by Shekhar Mishra
 - Helped by Shilpee Arora and Liz Clements.
 - Top 4 levels are controlled by Shekhar
 - Targeted at a broad audience.
- Pages useful for us can be linked from level 4 pages.
- Path to our pages:
 - Level 2: “Detector R&D”
 - Level 3: ”Physics and Simulation”
 - Level 4: “Simulated Analyses of Benchmark Processes”
 - Level 4: “Simulation and Reconstruction Software”
- Draft page (still in a very early stage):
 - ilc.fnal.gov/detector/rd/physics/technical/

Rob's Work

- Work with Hans and rest of SiD to define the missing infrastructure:
 - If it takes too long, we will give up and roll our own.
 - We know what we want.
- Learn/Port several codes:
 - Kalman filter
 - Vertex fitting
 - Jet finding
 - Flavor tagging
- $B(H \rightarrow b \bar{b})$ and $B(H \rightarrow c \bar{c})$.
- Help with overall design of our website.

Relevant Deadlines

- ALCPG October 22-26, 2007 at FNAL
 - First pass at one benchmark study for CDR.
- Spring 2008
 - Software for CDR benchmarks essentially complete.
 - CDR benchmark studies underway.
 - Start writing CDR.
- Fall 2008
 - Submit CDR.

Deadlines with Added Detail

- ALCPG October 22-26, 2007 at FNAL
 - Occupancy studies and most infrastructure done.
 - Kalman filter and other “ported” codes working
 - First release of detector built of wafers sometime in the summer.
 - First pass on one simulated analysis.
- Spring 2008
 - Our software working well enough for general use.
 - Continued improvements.
 - Several simulated analyses underway. TBD.
 - Start writing CDR.
- Fall 2008
 - Submit CDR.

Summary

- We have agreed to a list of jobs:
 - Lots of forward tracking.
 - One simulated analysis.
 - Precursor infrastructure work that is implied by this.
 - FNAL web site.
 - Simulation of Pb Glass-scint dual readout calorimeter.
- We have a rough outline of who is doing what with specific deadlines for the October ALCPG meeting and less specific details for afterwards.

Backup Slides

org.lcsim

- Java based.
- Can be run standalone or within JAS3.
 - Documentation/examples are JAS3-centric.
- Framework runs the event loop and executes a list of “drivers” specified by the user.
- Driver:
 - What other frameworks call a module.
 - Callable from the framework:
 - Detector change; process event; end of data ...
 - Can read event and add collections to the event.
 - Can overwrite/delete collections in an event.
- Native histogram/tuple environment: aida.
 - Display tools not as rich as root.

org.lcsim (2)

- Reconstruction code lives in user areas and is not vetted by anyone.
- Little discipline among users to ensure that their codes cooperate.
 - Predefined classes are not rich enough for the job.
 - So everyone makes their own private extensions.
 - Can add these objects to the event - but no persistency.
- No method to stop my histograms or collections from stomping on yours.
- Various “full” reconstruction codes are advertised:
 - Some ran in JAS2 and are not yet ported to JAS3.
 - Documentation by calling the author.
 - I have not yet run any of them.

org.lcsim (3)

- Release model
 - Infrequent releases.
 - Users: copy current .jar files from SLAC
 - Developers: build the head
 - You just gotta know when the head is/was in good shape.
- Each user keeps current .jar files in ~/.JAS3
 - Deploying a new release clobbers the old one and you cannot backtrack unless you saved a copy ahead of time or know the check out time of the old version.
- Presumes that you always have internet access.
 - It does cache things but you may need to know in advance if you need to force caching.